Maxmin-ω: A Simple Deterministic Asynchronous Cellular Automaton Scheme

Ebrahim L. Patel

Department of Mathematics, University of Portsmouth, Portsmouth PO1 3HF, UK
ebrahim.patel@port.ac.uk

Abstract. In this paper, we introduce the maxmin-ω system, a simple and intuitive model of asynchronous dynamics on a network. Each node in this system updates its state upon receiving a fixed proportion ω of inputs from neighbourhood nodes. We study the behaviour of nodal update times as a function of ω. Computational results suggest most complexity when ω is approximately 0.5. By implementing a cellular automaton (CA) under this maxmin-ω asynchronous scheme, we show some correspondence in complexity between timing and CA output. Moreover, our system can be interpreted by the useful modelling tool of max-min-plus algebra (MMP). We propose that the aforementioned results on complexity can be derived analytically via MMP.

1 Introduction

We introduce a simple asynchronous dynamical system, which we call maxmin-ω. The system acts on a network which, in this paper, will be a one-dimensional cellular automaton (CA) lattice (so the terms "nodes" and "cells" will be used interchangeably). There are a couple of crucial points that provide the attraction for studying this system: firstly, the update of cell states depends on local exchanges until the fraction ω is fulfilled; the maxmin-ω system is therefore deterministic, and is not only a departure from traditional asynchronous CA schemes (e.g., [1] and [2]) but differs from more recent work that looks at such local interactions that are stochastic [3]. Secondly, the parameter ω is shown to drive an interesting set of results – both in terms of the asymptotic timings and the implementation of a simple CA scheme – which leads us to suggest other, promising, applications other than CA.

2 The Maxmin-ω Model

In the maxmin-ω model a cell (node) state is updated at the end of a *cycle*. Consider a node i in a network of size N. The node carries a state that changes with time. Thus, we can plot points on the real line, representing time, corresponding to when these changes occur. We refer to the points as the *update times* of the nodal state. Let $x_i(k)$ denote the k^{th} update time for the i^{th} node.[1]. Once

[1] We choose x and not t as we will study update time as a 'state' itself; this is consistent with the literature [9]

each node in the neighbourhood of i has completed its k^{th} cycle (where k is also called a *cycle number*), it sends the updated state to i. The transmission of such a state from node j to i takes *transmission time* $\tau_{ij}(k)$. The update (or computation) of the state of node i takes a *processing time*, and it is represented in the k^{th} cycle by $\xi_i(k)$.

Now, suppose each node updates its state upon receiving a fraction ω of inputs from its neighbourhood ($\omega \in [0, 1]$). We define the "ω^{th} input" as the last of the fraction ω of inputs arriving at i. Then the $(k+1)^{\text{th}}$ update time of node i is given by the following recurrence relation.

$$x_i(k+1) = x_{(\omega)}(k) + \xi_i(k+1) \tag{1}$$

where $x_{(\omega)}(k)$ represents the k^{th} time of arrival of the ω^{th} input from the neighbourhood of i; if k is clear from context, we denote this $x_{(\omega)}$ for short. If there are n nodes in the neighbourhood of i, then $x_{(\omega)}$ practically represents the time of arrival of the m^{th} input where $m = \lceil \omega n \rceil$. Once node i receives the m inputs, it processes its new state; this takes time duration $\xi_i(k+1)$. Once processed, node i sends its state to downstream nodes at time $x_i(k+1)$, which is also the update time of i.

Figure 1 gives a flavour of the effect of maxmin-ω on CA using the same initial CA state and same CA rule. When $\omega = 1$ the CA space-time pattern resembles the synchronous CA pattern. This is because when $\omega = 1$, nodal states are updated upon arrival of *all* neighbourhood inputs, so there is no loss of information between the network states in the traditional synchronous model (where all nodes update at the same time) and this asynchronous system. On the other hand, the synchronous pattern is lost when $\omega < 1$.

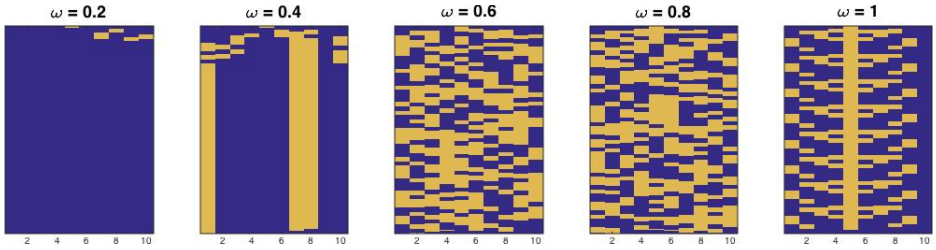

Fig. 1. CA space-time patterns as a function of ω for the maxmin-ω system. The underlying lattice comprises 10 cells. Each cell takes on one of two states, 1 and 0; state 1 is coloured light and state 0 is coloured dark.

2.1 Asymptotic Behaviour

Define the function $\mathcal{M}$ as the mapping $\mathcal{M} : \mathbb{R}^N \to \mathbb{R}^N$ whose components $\mathcal{M}_i$ are of the form of Equ. (1). We represent a system of N such equations by the

following.

$$\boldsymbol{x}(k+1) = \mathcal{M}(\boldsymbol{x}(k)) \tag{2}$$

for $k \geq 0$, where $\boldsymbol{x}(k) = (x_1(k), x_2(k), \ldots, x_N(k))$.

Denote by $\mathcal{M}^p(\boldsymbol{x})$ the action of applying $\mathcal{M}$ to a vector $\boldsymbol{x} \in \mathbb{R}^N$ a total of p times, i.e., $\mathcal{M}^p(\boldsymbol{x}) = \underbrace{\mathcal{M}(\mathcal{M}(\cdots(\mathcal{M}}_{p \text{ times}}(\boldsymbol{x}))\cdots))$.

Definition 1. *If it exists, the* cycletime vector *of $\mathcal{M}$ is $\chi(\mathcal{M})$ and is defined as* $\lim_{k\to\infty}(\mathcal{M}^k(\boldsymbol{x})/k)$.

Definition 2. *For some $k \geq 0$, consider the set of vectors*

$$\boldsymbol{x}(k), \boldsymbol{x}(k+1), \boldsymbol{x}(k+2), \ldots \in \mathbb{R}^N$$

where $\boldsymbol{x}(n) = \mathcal{M}^n(\boldsymbol{x}(0))$ for all $n \geq 0$. The set $x_i(k), x_i(k+1), x_i(k+2), \ldots$ is called a periodic regime *of $i \in \mathbb{N}$ if there exists $\mu_i \in \mathbb{R}$ and a finite number $\rho_i, \in \mathbb{N}$ such that*

$$x_i(k+\rho_i) = \mu_i + x_i(k).$$

The period *of the regime is ρ_i and $\chi_i = \mu_i/\rho_i$ is the* cycletime of i. *The smallest k for which the periodic regime exists is called the* transient time.

Under our initial conditions, K_i will be finite (see Theorem 1) and so, maxmin-ω always yields a periodic regime with the following system-wide quantities.

$$K = \max_i\{K_i\}, \quad \rho = \mathrm{LCM}_i(\rho_i), \quad \chi = (1/N)\sum_{i=1}^{N}\chi_i.$$

2.2 The One-Dimensional CA Network

We implement the maxmin-ω system on the one-dimensional (1D) CA lattice. This lattice has a natural definition of neighbourhood, i.e., the neighbourhood $\mathcal{N}_i$ of cell i of radius r is $\{i-r, \ldots, i-1, i, i+1, \ldots, i+r\}$ [6].[2]

From now on, we take $\xi_i(k)$ and $\tau_i(k)$ to be independent of k, so they are denoted ξ_i and τ_i, respectively. A study of the effect of r is beyond the scope of this short paper, suffice it to say that the results presented here are typical of those produced by most values of r (see [5], Ch. 5). We conduct three experiments with $N = 50$ and $r = 10$; they may best be described by the following.

Algorithm 1. *1. Choose $\xi_i, \tau_i \in \mathbb{Z}$ both from the uniform distribution (with equal probability) where $1 \leq \xi_i \leq \xi_{\max}$ and $1 \leq \tau_i \leq \tau_{\max}$.*
2. Taking an initial vector, $\boldsymbol{x}(0)$, run the maxmin-ω system for each ω value from 0.05 to 1, in steps of 0.05 (so there are 20 maxmin-ω systems to run).
3. For each maxmin-ω system, record the period ρ and cycletime χ.

[2] We take a finite lattice, so cells may be regarded as being arranged in a ring.

4. *Repeat above three steps 100 times to obtain, for each maxmin-ω system above, 100 independent periods and cycletimes.*
5. *For each maxmin-ω system, record the mean and median of the 100 periods and cycletimes obtained.*

We know that transient time K is always finite, therefore we don't exhibit transient time results here. We are most interested in the period ρ, which we take as a measure of the complexity of the system.

In experiment (i), we initialise to $\boldsymbol{x}(0) = \mathbf{0} = (0, 0, \ldots, 0)$ for all 100 runs; in experiment (ii), for each of the 100 runs, the elements of $\boldsymbol{x}(0) \in \mathbb{Z}$ are selected uniformly with equal probability where $0 < x_i(0) \leq 10$. In both of these experiments, we take $(\xi_{\max}, \tau_{\max}) = (10, 10)$. Experiment (iii) is a repeat of the second experiment but now taking $(\xi_{\max}, \tau_{\max}) = (20, 20)$. Figures 2 and 3 plot the results, a few notable features of which are as follows.

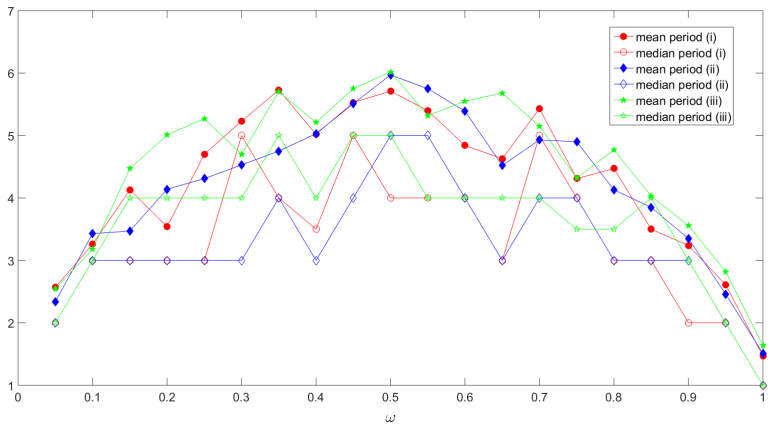

Fig. 2. Periods for the three experiments on the size 50 lattice with $r = 10$.

1. Under different initial conditions, the cycletime is almost identical. This is why we only alter the parameters ξ_i and τ_i in experiment (iii).
2. Increasing $(\xi_{\max}, \tau_{\max})$ values implies a larger cycletime; this fits with intuition since larger ξ_i and τ_i values would delay the processing of nodal states; subsequently the system takes longer to settle into some periodic behaviour.
3. The period curve is maximal when $\omega \approx \frac{1}{2}$. This forms part of a symmetrical curve, with smallest values at the two extremes of $\omega \approx 0$ and $\omega \approx 1$.

3 Cellular Automata in Maxmin-ω Time

Let $s_i(k)$ denote the (CA) state of a cell i at cycle $k \in \mathbb{N}$, so that the state of the system at cycle k is represented by the vector $\boldsymbol{s}(k) = (s_1(k), s_2(k), \ldots, s_N(k))$.

Suppose a cell is contained in a neighbourhood of size $2r+1$; then a CA rule is a function $f : \{0,1\}^{2r+1} \to \{0,1\}$ given by $s_i(k+1) = f(\mathcal{N}(s_i(k)))$, where $\mathcal{N}(s_i(k))$ denotes the CA states of $\mathcal{N}_i$ in cycle k. Note that the same cycle k does not imply the same real time $t \in \mathbb{R}$; this is due to asynchrony. We focus on the CA rule

$$s_i(k+1) = \sum_{\substack{j \in \mathcal{N}_i \\ x_j(k) + \tau_{ij} \leq x_{(\omega)}(k)}} s_j(k) \bmod 2 \tag{3}$$

i.e., the state of each cell is the sum (modulo 2) of the states of those neighbours of i that arrive before or at the same time as the ω^{th} input.

3.1 Classification

To numerically classify the CA space-time output as a function of ω, we use two measures in tandem, as provided by Marr and Hütt in [7]. The first measure is the *Shannon entropy* $S \in [0,1]$, which relies on the densities of CA states 0 and 1 in the time series of the evolving CA states of a cell. The second measure we employ is the *word entropy* $W \in \mathbb{R}_+$, which depends on the occurrence of blocks of constant states in the time series of a cell. CA space-time patterns can now be classified according to their S and W values; a large S or W value generally signifies large complexity of CA pattern.

3.2 CA Results

For each maxmin-ω system of Algorithm 1, we also implemented the CA rule of Equ. (3). That is, steps 3, 4, and 5 were extended to record Shannon and word entropies from the 100 runs. Here, we present these CA results.

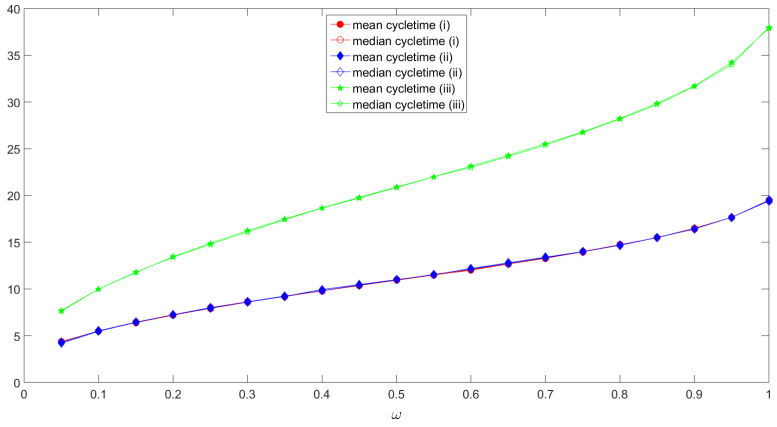

Fig. 3. Cycletimes for the three experiments on the size 50 lattice with $r = 10$.

Again, we generated results for the three experiments of Sect. 2.2. The initial CA state (when $k = 0$) was randomised for each of the 100 runs, and each CA was iterated 250 times. The corresponding entropy results are in Fig. 4.

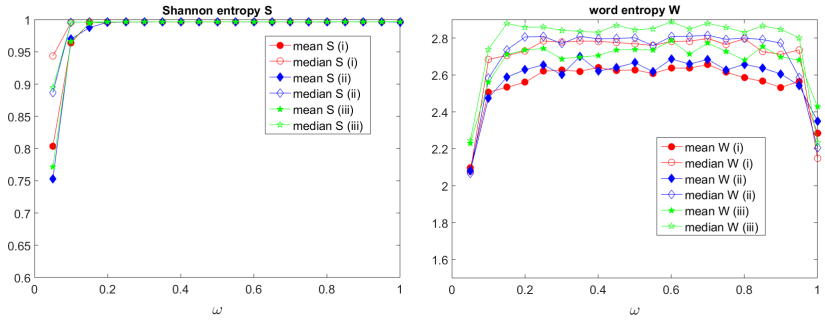

Fig. 4. CA entropies for the three experiments of Sect. 2.2.

4 Max-min-plus Algebra

The maxmin-ω system of Equ. (2) can be reinterpreted in terms of a curious branch of mathematics called *max-min-plus algebra* (MMP) [5]. We leave the details of MMP to the highly accessible expositions in [8] and [9]. What we must state here is that once maxmin-ω is represented as a max-min-plus function then transient time, period, and cycletime are readily understood concepts. The following theorem (see [9], Theorem 12.7 for proof) provides some significance.

Theorem 1. *If the cycletime vector $\boldsymbol{\chi}(\mathcal{M})$ of a max-min-plus function $\mathcal{M}$ exists for some finite vector $\boldsymbol{x}$ (i.e. where all elements of $\boldsymbol{x}$ are finite), then it exists for all finite vectors $\boldsymbol{x}$ and $\boldsymbol{\chi}(\mathcal{M})$ is independent of the initial condition $\boldsymbol{x}$.*

This theorem explains why our cycletime is invariant under different initial conditions (see Fig. 3). As for the periods being similar (except, perhaps, in experiment (iii)), this can be explained analytically by the "Duality Theorem" [10], a result that shows that the period is dependent on network structure and the parameters ξ_i and τ_i. MMP thus looks to be a promising way to study maxmin-ω analytically.

5 Discussion

We have shown that a simple, deterministic, asynchronous system produces intriguing results. In line with the periods, CA complexity for the simple rule we have employed appears maximal when ω is not near 0 or 1 (see Fig. 4). A natural extension is to study the effect on different CA rules. Moreover, while this work

focused on the 1D lattice, preliminary work suggests that it is also possible to obtain similar results for variable lattices/networks [5]. In particular, the maximal complexity at $\omega \approx 1/2$ seems universal, regardless of network type. This leads us to ask whether this 'middle system complexity' exists in applications such as neural networks and epidemic processes on a network.

The reason for the name "maxmin-ω" is now evident. A MMP representation is a first step towards analytically understanding the complexities of maxmin-ω and any associated information exchange system (such as virus transmission), not only CA. We have conjectured ways to make progress in this in [5].

References

1. B. Schönfisch and André de Roos, "Synchronous and asynchronous updating in cellular automata," *BioSystems*, **51**(3), 123–143, 1999.
2. N. Fatès and M. Morvan, "An Experimental Study of Robustness to Asynchronism for Elementary Cellular Automata," *Complex Systems*, **16**:1–27, 2005.
3. O. Bouré, N. Fatès, & V. Chevrier, "Probing robustness of cellular automata through variations of asynchronous updating," *Natural Computing*, **11**(4), 553–564, 2012.
4. E. L. Patel and D. Broomhead, "A Max-Plus Model of Asynchronous Cellular Automata," *Complex Systems*, **23**(4), 2014.
5. E. L. Patel, *Maxmin-plus models of asynchronous computation*, PhD thesis, University of Manchester, 2012.
6. S. Wolfram, "Statistical mechanics of cellular automata," *Reviews of Modern Physics*, **55**(3):601, 1983.
7. C. Marr and M.T. Hütt. "Topology regulates pattern formation capacity of binary cellular automata on graphs," *Physica A: Statistical Mechanics and its Applications*, **354**: 641–662, 2005.
8. F. Baccelli, G. Cohen, G. J. Olsder, & J-P. Quadrat, *Synchronization and linearity: an algebra for discrete event systems*. John Wiley & Sons Ltd, 1992.
9. B. Heidergott, G. J. Olsder, & J. W. van der Woude, *Max Plus at Work: Modeling and Analysis of Synchronized Systems: a Course on Max-Plus Algebra and its Applications*, Princeton Series in Applied Mathematics (**13**), Princeton University Press, 2006.
10. S. Gaubert and J. Gunawardena. "The duality theorem for min-max functions," *Comptes Rendus de l'Académie des Sciences-Series I-Mathematics*, **326**(1):43–48, 1998.